

A2
cm/ld

33. A computer-readable medium having computer-executable instructions for performing the steps recited in claim 31.

REMARKS

The specification has been amended for style. Claims 30 through 33 have been added to the claim set. Support for these new claims is found in the specification and figures as originally filed. No new matter has been added by way of these amendments.

Conclusion

The application is considered in good and proper form for allowance, and the Examiner is respectfully requested to pass this application to issue. If, in the opinion of the Examiner, a telephone conference would expedite the prosecution of the subject application, the Examiner is invited to call the undersigned attorney.

Respectfully submitted,



John B. Conklin, Reg. No. 30,369
One of the Attorneys for Applicant
LEYDIG, VOIT & MAYER, LTD.
Two Prudential Plaza, Suite 4900
180 North Stetson
Chicago, Illinois 60601-6780
(312) 616-5600 (telephone)
(312) 616-5700 (facsimile)

Date: January 15, 2002

APPENDIX A

PROPOSED CHANGES TO BE MADE TO THE SPECIFICATION UPON FILING OF PRELIMINARY AMENDMENT A

In accordance with the present invention, a[A] computer-executable application driver scans a GUI of an application to be explored for information related to its graphics elements. This results in the acquisition of information related to top-level graphics elements of the GUI. These elements characterize the initial state of the application. From this information, the application driver chooses one of the top-level elements and then chooses an action to perform on that element. The application driver records each state it encounters and each action it performs. For example, the application driver records its encounter with a particular state, [say a dialog box] such as the presentation of a particular dialog box, and records that it activated a button within the dialog box. The next time the same dialog is encountered, the application driver may choose the same button but may select a different action to be performed. In a preferred embodiment, [It]it will continue to select this button each time it encounters the dialog until all actions associated with the button have been performed. Whenever an action is performed, the record of elements and associated actions is updated.

APPENDIX B

CLEAN COPY OF THE ENTIRE SET OF PENDING CLAIMS, FOLLOWING ENTRY OF
PRELIMINARY AMENDMENT A

1. A method for generating a map that associates a graphics element of a graphical user interface of a software application with an executable feature of the software application, the method comprising:
 - retrieving information descriptive of the graphics element, the information including an executable feature associated with the graphics element;
 - storing the executable feature in association with the graphics element; and
 - executing an executable feature stored in association with a graphics element.
2. The method of claim 1 further comprising, in response to the executable feature exposing a second graphics element:
 - retrieving information descriptive of the second graphics element, the information including a second executable feature associated with the second graphics element;
 - storing the second executable feature in association with the second graphics element; and
 - executing the second executable feature stored in association with the second graphics element.
3. The method of claim 1 wherein the retrieving comprises capturing information pertaining to the graphics element.
4. The method of claim 1 wherein the storing includes updating an indicator associated with the graphics element when an executable feature stored in association with the graphics element is executed.
5. The method of claim 1 wherein the storing includes organizing the retrieved information such that an executable feature stored in association with the graphics

element can be interpreted by a computer-executable application capable of accessing the retrieved information.

6. The method of claim 1 wherein the storing includes organizing the retrieved information such that an executable feature stored in association with the graphics element can be interpreted by a user capable of accessing the retrieved information from memory.

7. The method of claim 1 wherein the executing comprises selecting from the stored information an executable feature stored in association with a graphics element.

8. The method of claim 7 wherein the selecting comprises selecting an executable feature not previously executed.

9. The method of claim 8 wherein the selecting comprises reviewing an indicator to select an executable feature not previously executed.

10. The method of claim 7 wherein the selecting comprises selecting executable features in a depth-first mode of operation.

11. The method of claim 7 wherein the selecting comprises selecting executable features in a breadth-first mode of operation.

12. A computer-readable medium having computer-executable instructions for performing the method recited in claim 1.

13. A system for generating a map that associates a graphics element of a graphical user interface of a software application with an executable feature of the software application, the system comprising:

a capture agent for retrieving information descriptive of the graphics element, the information including an executable feature associated with the graphics element;

an application driver for storing the executable feature in association with the graphics element; and

a command agent for executing an executable feature stored in association with a graphics element.

14. The system of claim 13 wherein the capture agent is invoked by the application driver.

15. The system of claim 13 wherein the capture agent submits the retrieved information to the application driver.

16. The system of claim 13 wherein the application driver selects a graphics element to be executed from the stored information.

17. The system of claim 16 wherein the application driver selects an executable feature not previously executed.

18. The system of claim 17 wherein the application driver reviews an indicator to select an executable feature not previously executed.

19. The system of claim 16 wherein the application driver selects executable features in a depth-first mode of operation.

20. The system of claim 16 wherein the application driver selects executable features in a breadth-first mode of operation.

21. The system of claim 13 wherein the application driver updates an indicator associated with the graphics element when an executable feature stored in association with the graphics element is executed.

22. A method for systematically invoking an executable feature of a software application having a graphical user interface, the graphical user interface displaying a graphics element associated with the executable feature, the method comprising:

 retrieving information descriptive of the graphics element, the information including an executable feature associated with the graphics element;

 storing the executable feature in association with the graphics element;

 selecting from the stored information an executable feature not previously executed; and

 executing the selected executable feature.

23. The method of claim 22 further comprising, in response to the executable feature exposing a second graphics element:

 retrieving information descriptive of the second graphics element, the information including a second executable feature associated with the second graphics element;

 storing the second executable feature in association with the second graphics element; and

 selecting from the stored information a second executable feature not previously executed; and

 executing the selected second executable feature.

24. The method of claim 22 wherein the retrieving comprises capturing information pertaining to the graphics element.

25. The method of claim 22 wherein the storing comprises updating an indicator associated with the graphics element when an executable feature stored in association with the graphics element is executed.

26. The method of claim 22 wherein the selecting comprises reviewing an indicator to determine an executable feature not previously executed.

27. The method of claim 22 wherein the selecting comprises selecting executable features in a depth-first mode of operation.
28. The method of claim 22 wherein the selecting comprises selecting executable features in a breadth-first mode of operation.
29. A computer-readable medium having computer-executable instructions for performing the steps recited in claim 22.
30. The method of claim 1 wherein the graphical user interface is generated by a software application included in the set: an application program, an operating system, and a program module.
31. A method for generating a map that associates a graphics element of a graphical user interface with an executable feature, the method comprising:
 - retrieving information descriptive of the graphics element, the information including an executable feature associated with the graphics element;
 - storing the executable feature in association with the graphics element; and
 - executing an executable feature stored in association with a graphics element.
32. The method of claim 31 further comprising, in response to the executable feature exposing a second graphics element:
 - retrieving information descriptive of the second graphics element, the information including a second executable feature associated with the second graphics element;
 - storing the second executable feature in association with the second graphics element; and
 - executing the second executable feature stored in association with the second graphics element.
33. A computer-readable medium having computer-executable instructions for performing the steps recited in claim 31.